Python for Biologists - Local Development Setup

Mac & Linux Installation Guide

Course: Year 3 Biology Python Programming **Instructor**: Helfrid Hochegger (hh65@sussex.ac.uk) **Last Updated**: 2025-01-14

Overview

This guide will walk you through setting up a professional Python development environment on your Mac or Linux machine. Follow these steps in order!

What you'll install:

- ✅ Warp Terminal (modern terminal with AI features)
- ✅ Git (for downloading course materials)
- ✅ UV (Python package manager)
- ✅ Cursor IDE (code editor with AI assistance)
- ✅ Course notebooks repository

Estimated Time: 30-45 minutes

Step 1: Install Warp Terminal

Warp is a modern terminal with AI features that makes command-line work much easier for beginners.

Mac Users

1. Visit warp.dev
2. Click "Download for Mac"
3. Open the downloaded .dmg file
4. Drag Warp to your Applications folder
5. Open Warp from Applications
6. Sign in with your email (free account)

Linux Users

1. Visit warp.dev
2. Download for Linux
3. Follow the installation instructions for your distribution
4. Launch Warp

Alternative: If you prefer, you can use your system's default terminal (Terminal on Mac, GNOME Terminal on Linux). All commands work the same!

Step 2: Check if Git is Installed

Before installing anything, let's check if you already have Git.

Open Warp (or your terminal) and run:

```
git --version
```

If you see a version number (like `git version 2.39.0`):

- ✅ **You're good to go!** Skip to Step 3.

If you see "command not found":

- Continue with the installation instructions below.

Mac Users: Install Git via Homebrew

Option 1: Homebrew (Recommended)

First, install Homebrew (a package manager for Mac):

```
/bin/bash -c "$(curl -fsSL https://raw.githubusercontent.com/
        Homebrew/install/HEAD/install.sh)"
```

Follow the on-screen instructions and enter your password when prompted.

Then install Git:

```
brew install git
```

Option 2: Xcode Command Line Tools

If you run `git --version` and get a popup asking to install developer tools, click "Install". This is Apple's built-in option (but Homebrew is better long-term).

Linux Users: Install Git

Ubuntu/Debian:

```
sudo apt update
sudo apt install git -y
```

Fedora:

```
sudo dnf install git -y
```

Verify Git Installation

```
git --version
```

You should now see a version number!

Step 3: Clone the Course Repository

Now let’s download all the course materials from GitHub.

Navigate to where you want to store your course files:

```
cd Documents
mkdir biology-python
cd biology-python
```

Clone the repository:

```
git clone https://github.com/HocheggerLab/y3-bio-python.git
```

You’ll see output like:

```
Cloning into 'y3-bio-python'...
remote: Enumerating objects...
Receiving objects: 100% done.
```

Navigate into the repository:

```
cd y3-bio-python
```

✅ **Success!** You now have all the course notebooks on your computer.

Step 4: Install UV (Python Package Manager)

UV is a modern, fast Python package manager that will handle all your dependencies.

Install UV:

Mac & Linux:

```
curl -LsSf https://astral.sh/uv/install.sh | sh
```

Restart your terminal

Close Warp and open it again (or open a new terminal tab).

Verify installation:

```
uv --version
```

You should see the UV version number!

Step 5: Set Up Python Environment

Now let’s install all the course packages (numpy, pandas, matplotlib, jupyter, etc.)

Make sure you’re in the course repository folder:

```
cd ~/Documents/biology-python/y3-bio-python
```

(Adjust the path if you put it somewhere else)

Sync all course packages:

```
uv sync
```

This will:

- Read the `pyproject.toml` file
- Create a virtual environment (`.venv` folder)
- Install Python 3.12 and all course packages

This takes 2-3 minutes. You’ll see output like:

```
Resolved 23 packages in 450ms
Installed 23 packages in 2.1s
```

✅ **Done!** Your Python environment is ready.

Step 6: Download and Install Cursor IDE

Cursor is like VS Code but with AI superpowers - perfect for learning Python!

Download Cursor:

1. Visit cursor.com
2. Click “Download for Mac” or “Download for Linux”
3. **Mac**: Open the .dmg file and drag Cursor to Applications
4. **Linux**: Follow the installation instructions for your distribution

Launch Cursor:

1. Open Cursor from Applications (Mac) or your app menu (Linux)
2. Grant any permissions if prompted

Sign in (optional but recommended):

- Click “Sign In” in the bottom-left
- Create a free account or sign in with GitHub
- This enables AI features

Step 7: Open the Repository in Cursor

Option 1: From Terminal

In your terminal (make sure you’re in the `y3-bio-python` folder):

```
cursor .
```

This opens the current folder in Cursor!

Option 2: From Cursor

1. Open Cursor
2. File → Open Folder
3. Navigate to `~/Documents/biology-python/y3-bio-python`
4. Click “Open”

Step 8: Install Required Cursor Extensions

You need two extensions to work with Jupyter notebooks.

Install Python Extension:

1. Click the Extensions icon in the left sidebar (looks like building blocks)
    - Or press `Cmd+Shift+X`
2. Search for “Python”
3. Install the one by **Microsoft** (it has the most downloads)

Install Jupyter Extension:

1. In the Extensions sidebar, search for "Jupyter"
2. Install the one by **Microsoft**
3. **Restart Cursor** after installation

Note: Cursor usually prompts you to install these automatically when you open a .ipynb file!

Step 9: Select Python Interpreter

Tell Cursor to use your virtual environment.

Steps:

1. Press `Cmd+Shift+P` to open the command palette
2. Type "Python: Select Interpreter"
3. Select the interpreter that shows `.venv` in the path
   - Should look like: `Python 3.12.x ('.venv': venv)`

You only need to do this once per project!

Step 10: Test Your Setup

Let's make sure everything works by running a notebook!

Open a notebook:

1. In the Cursor file explorer (left sidebar), navigate to:

   ```
   notebooks/lecture_1/
   ```

2. Click on `L1_N1_collab_notebooks.ipynb`

Run the first cell:

1. Look for the first code cell in the notebook
2. Click the ▶️ play button next to it
   - Or press `Shift+Enter`
3. Wait a moment for the kernel to start

What should happen:

- The cell runs
- You see output below the cell
- A number appears in brackets like `[1]` next to the cell

If it asks you to select a kernel:

1. Click “Select Kernel” at the top-right of the notebook
2. Choose “Python Environments…”
3. Select the one with `.venv` in the name

✅ You’re All Set!

Congratulations! Your development environment is fully set up. You can now:

- ✅ Open any notebook in the `notebooks/` folder
- ✅ Run cells with `Shift+Enter`
- ✅ Ask Cursor AI for help (press `Cmd+L`)
- ✅ Work offline with all course materials

Quick Reference Commands

```
# Navigate to your project
cd ~/Documents/biology-python/y3-bio-python

# Open in Cursor
cursor .

# Get latest course updates
git pull

# Add new Python packages (if needed)
uv add package-name
```

Troubleshooting

“git: command not found”

- Make sure you completed Step 2
- On Mac, if Homebrew is installed: `brew install git`

“uv: command not found”

- Restart your terminal after installing UV
- Make sure the install script completed successfully

Notebook won’t run / “No kernel”

- Make sure you selected the `.venv` interpreter (Step 9)

- Try restarting Cursor
- Make sure both Python and Jupyter extensions are installed

“Cannot find module ‘numpy’” or similar

- Make sure you ran `uv sync` in Step 5
- Make sure you selected the correct interpreter (should show `.venv`)

Extension installation fails

- Check your internet connection
- Try restarting Cursor and installing again
- Extensions sidebar: click the “…” menu → “Check for Extension Updates”

Getting Help

- **Cursor AI**: Press `Cmd+L` and ask questions about your code
- **Course Seminars**: Bring setup questions to practical sessions
- **Course Forums**: Post questions - other students can help!
- **Instructor**: Email hh65@sussex.ac.uk for technical issues

---

What’s Next?

1. **Explore the notebooks**: Start with `notebooks/lecture_1/`
2. **Practice running cells**: Get comfortable with the interface
3. **Try Cursor AI**: Press `Cmd+L` and ask it to explain Python concepts
4. **Attend seminars**: We’ll work through exercises together!

Remember: Focus on learning Python, not mastering the IDE. The tools are here to help you code!

Happy coding! 🐍🔬